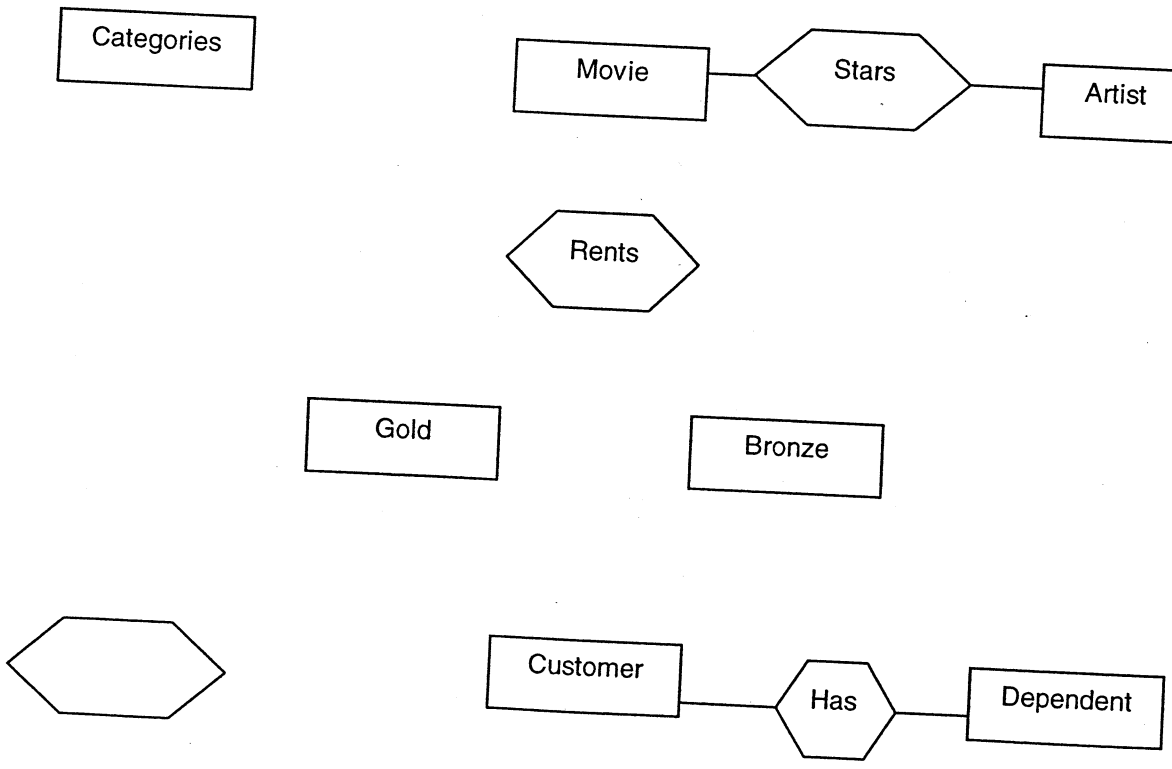
General Guidelines: *Exam duration: 50 min. Closed book, no collaboration. No questions during the exam, if you are unsure, state your assumptions clearly. All questions need to be answered. Use the spaces after the questions in this set of pages for your answers (additional pages can be used, but should not be necessary). Marked exams will be available on March 01 (in class). Deadline for appeals is March 15 at 4pm.*

Question 1. Consider the following problem specification regarding a database you are required to design for a single video rental store:

- The store has several clients, for whom it knows their names and phone numbers (which are assumed to be unique among the customers). There are two kinds of customer, those with a credit card number on file (gold customers), and who can rent more than one movie at a time and those who do not have such information on file (bronze customers) and can rent only one movie at a time. Each customer may have a set of dependents, with known names, which are allowed to rent movies, one at a time, under his/her responsibility.
- Each movie has a title and is identified by a unique movie number. However, there may be more than one copy of each movie carried by the store. In addition, each movie has to belong to one of a given set of categories.
- For the purpose of advertisements, each customer may inform a favorite movie category.

(a) (25 points) Finalize the following ER model, using the notation seen in class, in such a way that it reflects as accurately as possible the above scenario. You may need to add, and/or remove, and/or modify entities, and/or relationships, and/or attributes, and/or cardinalities, etc.



(b) (5 points) Map to the relational model *only the relationships* of the ER-model you designed in (a). Note that even if your ER-model is incorrect, but your mapping is correct you will get credit for it.

Question 2 (5 points each – no partial marks)

Consider the following relations and definitions:

Employee(empno, empname, empcity, empsalary) – each tuple defines an employee called empname and identified by empno who lives in empcity and earns empsalary annually.

Department(deptno, depname, deptomgr) – each tuple represents a department identified by deptno that is named deptname, and who has a manager identified by deptmgr, which is a foreign key to empno in relation Employee.

Allocated(empno, deptno) – This relation host tuples representation that employee empno works in deptno; empno and deptno are foreign keys to empno and deptno in relations Employee and Department respectively.

DeptLocation(deptno, deptcity) – Each tuple in this relation represents the fact that department deptno (a foreign key to deptno in Department) is located in city deptcity.

Project(pname, pno, deptno) – The tuples in this relation denote the fact that each project named pname and identified by pno is being developed at deptno (foreign key to depno in Department).

Write, if possible, the following queries using SQL (use the notation seen in class):

a) Find the names of all employees who live in Edmonton

b) Retrieve the names of all employees who work for department 01

c) Retrieve all information about all employees who work for the "research' department

d) Retrieve all cities, with no repetition, such that there is a project being developed in a department whose manager is from the same city where the department is located

e) Find the numbers of all departments in which there are no projects being developed

f) Obtain the names of all projects whose managers earn more than the average salary of all managers

g) Retrieve the name of the employees who are not managers and who work on all projects developed by department 05

h) Show the difference between the maximum and minimum salaries, over all salaries which are greater than the average salary

i) Create a VIEW called ImportantCities that shows all cities that host more than 2 departments

j) Find the numbers of the departments with more than 10 employees

Question 3 (4 points – no partial marks) Using the same relations given in Question 2, "translate" the following SQL queries into English or explain why they are illegal:

a)
```
SELECT empname
FROM Employee
WHERE empsalary > ANY
        (SELECT empsalary
         FROM Employee, Department
         WHERE E.empno = D.deptmgr)
```

b)
```
SELECT empno, empname
FROM Employee
GROUP BY empcity
HAVING COUNT(*) > 10
```

c)
```
SELECT empname
FROM Employee
WHERE NOT EXISTS
        (SELECT *
        FROM Department
        WHERE deptomgr = empno)
```

d)
```
CREATE VIEW TempTable AS
        SELECT deptno
        FROM Allocated
        GROUP  BY deptno
        HAVING COUNT(*) > 5
```

e)
```
UPDATE Employee
SET empsalary = empsalary+1000
WHERE empno IN
        (SELECT empno
         FROM Employee E, Department D
         WHERE E.empno = D.deptomgr)
```